Discussion of Prior Art

The combination of the Rassen et al. reference with the Cochrane et al. reference fails to teach or suggest all of Applicants' claimed elements. The Rassen et al. reference merely discloses a method for automatically defining a query interface for a datamart. The datamart includes fact and dimension tables. The method comprises accessing a schema description and a query interface description for the datamart. The schema description specifies a schema, which in turn, defines the relationships between the fact tables and dimension tables of the datamart. The query interface description specifies the fields, related to the schema description that can be used in a query and the way in which results are to be presented to the user. The fields correspond to columns and rows in the fact tables. The schema description is used to create a first set of commands to create and populate the fact and dimension tables. Additionally, a second set of commands to create the query interface is created. Some commands of the first set of commands are executed causing the creation and population of the tables. Some commands of the second set of commands are executed causing the creation of a user interface. A query is generated using the user interface. The query is sent to the system for processing. The results of the query are presented to the user according to the second set of commands. (Rassen et al. at col. 2, lines 65-67 and col. 3, lines 1-18.)

Generally, the system allows a consultant to define a well-formed datamart. The system includes tables and columns that conform to the definition of the datamart. The system also includes additional columns for foreign key tracking, source system key mapping, and time and date tracking. The system has automatic indexing. The system enforces typing information about the data stored in the datamart. These additional features cause the datamart to operate in a consistent manner. One benefit of such consistent operation is that results are consistent in meaning from query to query. (Rassen et al. at col. 4, lines 48-59).

The Examiner has cited portions of the Rassen et al. reference in the rejection of Applicant's independent claims 1, 7, and 10. Specifically, the Examiner cites that the query/reporting program 104 supports the querying of the datamart 150 and presents

results of those queries. The query and reporting process 104 uses the measurement information 168 and the query and reporting information 169, in addition to the schema definitions 161, to query the datamart 150 and provide that information to the web server 186. The query/reporting information 169 includes filters and form definitions. The filters allow the user to filter different fields out of the datamart 150. The forms allow the users to indicate which fields a user is particularly interested in. (Rassen et al. col. 8, lines 43-53).

The Rassen et al. reference is silent with respect to various of Applicants' claimed elements including *inter alia,* examining the joining query and providing an aliasing list indicating at least one field from at least one dimension table indicated in the joining query and also indicating the identity of the at least one dimension table, the aliasing list thereby providing a list of to-be-aliased fields and corresponding dimension table identities. The Rassen et al. reference also fails to suggest or disclose providing an alias table for the at least one field in the aliasing list, the alias table including each value of the field occurring in the at least one dimension table and also including an alias value for each value of the field, and using as the alias value the value of the key field relating the dimension table to the hub table, the alias table thereby providing a table of alias field values and corresponding aliased field values. Still further, the Rassen et al. reference fails to teach Applicants' claimed step of transforming the joining query into a reduced query in which the aliased field values are replaced by the alias values. See e.g., Applicants' claims 1 and 7. The Rassen et al. reference also fails to teach or suggest a method including providing aliases instead of actual values for leaf nodes of the relational database, and selecting all the aliases for the leaf node from a fact table instead of an individual dimension table, thereby reducing the requirement for joins in said query statement, as set forth in Applicants' claim 10.

The Examiner acknowledges the shortfalls of the Rassen et al. reference noting that Rassen et al. does not clearly disclose the step of "examining the joining query and providing an aliasing list."

The teachings of the Rassen et al. reference cited by the Examiner does not meet the limitations Applicants' claims because the Rassen et al. reference does not teach or suggest e.g., that the relational database manager selects alias from the hub table instead

of actual field values from individual dimension tables, internally, i.e. before providing the actual response. Additionally, the Rassen et al. reference does not teach or disclose that the relational database manager replaces the aliases with the values of the aliased fields by referring to the alias tables, when providing the query as an output, e.g., see Applicants' specification at page 12, lines 11-20. The result is the same for a reduced SQL as for a joining SQL, but the reduction in processing is substantial because the reduced SQL does not require that the relational database manager perform any joins, e.g., see Applicants' specification at page 12, lines 16-20.

Further, the combination of the Cochrane et al. reference with the Rassen et al. reference does not remedy the defects of the Rassen et al. reference taken alone.

The Cochrane et al. reference merely teaches a system, method, and article of manufacture for supporting summary tables in a database system that does not otherwise support summary tables. The system generally comprises a central program and one or more database systems that may be heterogeneous. At least one of the database systems does not support the generation, maintenance, and/or querying of summary tables. The central program is configured to communicate with the database systems and to identify relations corresponding to summary tables (also referred to as materialized views) within one or more of the database systems. The central program may initiate the generation of summary tables, which may be populated local to the central program or local to one or more of the database systems. The central program may also maintain or coordinate maintenance of the summary tables. In addition, the central program is preferably configured to receive user queries on one or more of the database systems and to generate optimized query plans based upon the user queries, considering in so doing, the contents of the summary tables (Cochrane et al. at Abstract). The Cochrane et al. reference is silent with respect to the claimed elements of examining the joining query and providing an aliasing list indicating at least one field from at least one dimension table indicated in the joining query and also indicating the identity of the at least one dimension table, the aliasing list thereby providing a list of to-be-aliased fields and corresponding dimension table identities. The Cochrane et al. reference also fails to disclose providing an alias table for the at least one field in the aliasing list, the alias table including each value of

the field occurring in the at least one dimension table and also including an alias value for each value of the field, and using as the alias value the value of the key field relating the dimension table to the hub table, the alias table thereby providing a table of alias field values and corresponding aliased field values, and transforming the joining query into a reduced query in which the aliased field values are replaced by the alias values, as set forth in Applicants' claim 1.

The Examiner cites to specific portions of the Cochrane et al. reference in support of the rejection. Specifically, the Examiner cites FIG. 6 element 176, in the flow chart 170 that describes one method for responding to the query using summary tables. The step 176 consists of the query processing module parsing the query into initial data structures suitable for further operations.

The Examiner cites the Cochrane et al. reference at col. 8 lines 43-53, and col. 17 at lines 1-10. The Cochrane et al. reference teaches that when the summary table creation module 70 initiates the generation of a summary table 80, 68, it preferably instructs the generating entity, whether the central program 52 or a database manager 60, to assign the summary table 68, 80 an alias or "nickname" 90. The nickname 90 is preferably transmitted to the catalog 76 and stored there with other information 88 regarding the contents of the database 58. This information 88 preferably includes lists of tables within each database 58, the nature of the data stored in the tables, and the particular schema used in each table. (Cochrane et al. at col. 8, lines 43-53). The Cochrane et al. reference teaches examples of an arrangement that includes the situation where join indexes are used to create the summary table 80 and the database system 54 does not allow access to the join indexes.

The summary table is created within the remote database system 54 and is maintained by the remote database system 54. No routing is done by the central program 52, other than using this information for cost modeling purposes to choose a query plan. (Cochrane et al. at col. 16, lines 62-66, col. 17, lines 12-17).

The Examiner also cites the Cochrane et al. reference with regard to transforming the joining query into a reduced query, citing "If this query is sent directly to the database systems that do not support summary tables, the database systems will not be able to

make use of the materialized view RICHDEPT_EMP to optimize the query execution. The central program 52 (preferably through the query processing module 74), on the other hand, understands how to optimize the query using summary tables and transforms the query, preferably within the query plan 86." (Cochrane et al. at col. 13, lines 36-44).

As is clear from the above, the Cochrane et al. reference teaches summary tables, which are not the same as the claimed method. A summary table is a table whose definition is based on the result of a query. As such, the summary table typically contains results based on the data existing in the table or tables that its definition is based on. If the SQL compiler determines that a query will run more efficiently against a summary table than the base table, the query executes against the summary table, and a result is obtained faster.

Such summary tables are relational database specific terms and are not related to the elements in Applicants' claims. The Cochrane et al. reference teaches supporting summary tables in database systems that do not have support for summary tables. Applicants' claimed invention, however, does not utilize summary tables. Applicants' claimed method for extracting data depends upon the availability of the claimed aliases. The query generated in Applicants' claimed invention is generated after optimization and may not be possible using the teachings or suggestions of the Cochrane et al. reference.

With regard to the rejection of claims 5 and 6, Applicants' respectfully submit that the Rassen et al. reference does not teach the provision of the query comprising a select clause in which a field is selected from one of the dimension tables using an alias, the alias indicating, by way of an alias table, the value of the selected field in the dimension table as claimed.

According to the present invention, in order to avoid having to join to the hub table, each of the dimension tables referred to in a joining SQL statement, alias tables are created. An alias table for a dimension table contains an alias (the key field value in the hub table used to index into the dimension table) for each value of the field of the dimension table referred to in the joining SQL statement, e.g., as disclosed in Applicants' specification at page 11, lines 1-10. The query log as cited by the Examiner and illustrated in the Rassen et al. reference at column 41 merely illustrates that an aggregate

and navigation process determined which aggregate would be most appropriate for a query that was executed against the prior art datamart of Rassen et al.

In view of the above, independent claims 1, 5, 7 and 10, and each of the claims dependent thereon, are believed to be in condition for immediate allowance.

Further remarks regarding the asserted relationship between Applicants' claims and the prior art are not deemed necessary, in view of the above discussion. Applicants' silence as to any of the Examiner's comments is not indicative of an acquiescence to the stated grounds of rejection.

Conclusion

The Examiner is respectfully requested to reconsider this application, allow each of the pending claims and to pass this application on to an early issue. If there are any remaining issues that need to be addressed in order to place this application into condition for allowance, the Examiner is requested to telephone Applicants' undersigned attorney.

Respectfully submitted,

Andrew D. Gathy
Attorney for Applicant(s)
Registration No.: 46,441
Law Office of Barry R. Lipsitz
755 Main Street
Monroe, CT 06468
(203) 459-0200

ATTORNEY DOCKET NO.: IMRS-364
Date: August 20, 2004